

Sequential Complexity as a Descriptor for Musical Similarity*

Peter Foster, Matthias Mauch, and Simon Dixon[†]

March 3, 2014

Abstract

We propose string compressibility as a descriptor of temporal structure in audio, for the purpose of determining musical similarity. Our descriptors are based on computing track-wise compression rates of quantised audio features, using multiple temporal resolutions and quantisation granularities.

To verify that our descriptors capture musically relevant information, we incorporate our descriptors into similarity rating prediction and song year prediction tasks. We base our evaluation on a dataset of 15 500 track excerpts of Western popular music, for which we obtain 7 800 web-sourced pairwise similarity ratings. To assess the agreement among similarity ratings, we perform an evaluation under controlled conditions, obtaining a rank correlation of 0.33 between intersected sets of ratings. Combined with bag-of-features descriptors, we obtain performance gains of 18.5% and 9.9% for similarity rating prediction and song year prediction. For both tasks, analysis of selected descriptors reveals that representing features at multiple time scales benefits prediction accuracy.

1 Introduction

We are concerned with the task of quantifying musical similarity, which has received considerable interest in the field of audio-based music content analysis [1, 2]. Owing to the proliferation of music in digital formats and the expansion of web-based music databases, there is an impetus to develop novel search, navigation and recommendation systems. Music content analysis has found application in such information retrieval systems as an alternative to manual annotation processes, when the latter are infeasible, unavailable or amenable to be supplemented [3].

*This work has been submitted to the IEEE for possible publication. Copyright may be transferred without notice, after which this version may no longer be accessible.

[†]All authors are with the School of Electronic Engineering and Computer Science, Queen Mary University of London, London, E1 4NS, UK. (Email: peter.foster@eecs.qmul.ac.uk; matthias.mauch@eecs.qmul.ac.uk; simon.dixon@eecs.qmul.ac.uk)

We may distinguish between music content analysis applications such as audio fingerprinting [4], version identification [5], genre classification [6] and mood identification [7]. Given a query track, audio fingerprinting typically should identify a unique track deemed similar with respect to a collection. In contrast, for genre and mood classification, the set of tracks deemed similar with respect to a collection is typically large. Correspondingly, we may distinguish between music classification tasks according to the degree of *specificity* associated with the measure of musical similarity [1].

In this work, we consider two low-specificity tasks, namely similarity rating prediction and song year prediction. An important issue in our considered domain surrounds feature representation. In particular, we address the problem of representing temporal structure in audio features. We refer to summary statistics of audio features extracted from a song as descriptors. Descriptors may be characterised according to how temporal structure is accounted for [2]. We may distinguish between *bag-of-features* representations [8], which discard information on temporal structure, and sequential representations. As a sequential representation, we propose to estimate the complexity of audio feature time series, where we quantify complexity in terms of string compressibility. As a result, we obtain scalar-valued summary statistics which retain information on temporal structure.

Our evaluations involving similarity rating prediction and song year prediction test the hypothesis that our complexity descriptors capture temporal information in audio features and that such information is relevant for determining musical similarity. For similarity rating prediction, our ground truth is given by human similarity judgements and we assume that an objective musical similarity correlates with subjects' degree of perceived musical similarity, based on a five-point rating scale. For song year prediction, our ground truth is readily given by chart entry times of songs and we assume that musical similarity correlates with chart entry time proximity.

Section 2 provides an overview of methods and descriptors for computing low-specificity similarity. In Section 3, we describe our approach. In Section 4, we detail our experimental method and results; we provide separate accounts for similarity rating prediction and song year prediction in Sections 4.1 and 4.2, respectively. Finally, in Section 5 we provide conclusions.

2 Background

For a detailed review of recent literature on methods for determining musical similarity, from the perspective of classification, we refer to the work of Fu et al. [2]. To determine musical similarity, one possible approach involves computing pairwise distances between tracks. The obtained distances may then be used for classification. A second approach consists in applying track-wise descriptors directly for classification.

Based on the second approach, Tzanetakis and Cook [9] compute first and second-order moments on spectral features including MFCCs, to perform genre

classification using the k -nearest neighbours (KNN) algorithm and Gaussian mixture models (GMMs) estimated on each target class. Li and Ogihara [10] propose to classify Daubechies wavelet histograms using GMMs and KNN for genre and mood classification. Using spectral features, West et al. [11] propose methods for learning similarity functions based on constructing decision trees for genre classification. Slaney et al. [12] propose feature transformations based on supervised learning and using onset and loudness features, for the purpose of album and artist classification.

Based on the approach of determining distances between descriptors, Logan and Salomon [13] propose to estimate GMMs on individual tracks. Pairwise track distances are then computed using a combination of Kullback-Leibler divergence (KLD) and earth mover’s distance, where the KLD is used to compare pairs of track centroids. The approach based on KLD assumes that each centroid follows a Gaussian distribution; correspondingly the KLD may be computed in closed form as

$$\text{KLD} = \frac{1}{2} \left(\text{tr}(\Sigma_1^{-1} \Sigma_2) + (\mu_1 - \mu_2)^T \Sigma_1^{-1} (\mu_1 - \mu_2) - h - \log \frac{|\Sigma_2|}{|\Sigma_1|} \right) \quad (1)$$

where Σ_1, Σ_2 and μ_1, μ_2 respectively denote the mean and covariance of two multivariate Gaussian distributions with dimensionality h . Aucouturier and Pachet [14] in contrast compute cross-likelihoods between GMMs using Monte Carlo approximations for the purpose of genre classification, whereas Berenzweig et al. [15] consider the asymptotic likelihood approximation of the KLD and centroid distances for the task of similarity rating prediction. Mandel and Ellis [16] instead represent tracks as single Gaussians and use (1) as a distance measure between track pairs. The obtained distances are then applied to artist identification, using support vector machines (SVMs) for classification. An alternative approach to computing the KLD is based on computing histograms of quantised features, as proposed by Vignoli and Pauws [17] for playlist recommendation; Levy and Sandler [18] compare approaches in the context of genre classification.

The previously described techniques are commonly referred to *bag-of-features* approaches, since they discard information on temporal structure. Yet, the relative convenience of bag-of-features approaches stands in contrast to the importance of temporal structure in perception of musical timbre, as observed by McAdams et al. [19]. Correspondingly, Aucouturier and Pachet [8] argue that the bag-of-features approach is insufficient to model polyphonic music for determining similarity. Sequential representations based on mid-level features are widely applied for the purpose of version identification [5]. For low-specificity classification, one possible approach to mitigating the shortcoming of the *bag-of-features* approach involves the intermediate step of aggregating features locally, before summarising anew using obtained summary statistics. Tzanetakis and Cook [9] propose to estimate the local mean and variance of features contained in

a 1s window. For the task of predicting musical similarity, Seyerlehner et al. [20] apply a single, global summarisation step to overlapping windows, computing variance and percentiles. For the purpose of local aggregation, alternative pooling functions are considered by Mörchen et al. [21], Hamel et al. [22], Wülfing and Riedmiller [23].

An alternative approach is based on retaining the temporal order of features at each window position. Spectral analysis may be applied to the original features, resulting in a new feature sequence. Pampalk [24] proposes fluctuation patterns describing loudness modulation across frequency bands, whereas Lee et al. [25] propose statistics based on modulation spectral analysis. Mörchen et al. [21] consider a variety of statistics based on spectral analysis and autocorrelation. Meng et al. [26], Coviello et al. [27] apply multivariate autoregressive modelling to windowed features, for the tasks of genre and tag classification.

To account for temporal structure, statistical modelling may be applied to quantised features. For genre classification, Li and Sleep [28] propose an SVM kernel in which pairwise distances are obtained by comparing dictionaries generated using the Lempel-Ziv compression algorithm [29]. Reed and Lee [30] apply latent semantic analysis to unigram and bigram counts for classification using SVMs, whereas Langlois and Marques [31] propose to estimate language models for computing sequence cross-likelihoods for genre and artist classification. Ren and Jang [32] propose an algorithm for computing histograms of feature codeword sequences for genre classification.

Recent approaches attempt to model temporal structure using representations constructed at multiple time scales. Based on a bag-of-features approach, Foucard et al. [33] propose an ensemble of classifiers, where each classifier is trained on features at a given time scale. Features at successive resolutions are aggregated using averaging. Applied to tag and instrument classification, results indicate that a multiscale approach benefits performance. Dieleman and Schrauwen [34] apply feature learning based on spherical k -means clustering to tag classification. Evaluated aggregation techniques are based on varying the spectrogram window size, in addition to Gaussian and Laplacian pyramid smoothing techniques. Although not applied to classification, Mauch and Levy [35] propose a similar smoothing approach for characterising structural change at multiple time scales. Finally, convolutional neural networks have been proposed for modelling temporal structure: Dieleman et al. [36] propose deep learning architectures for genre, artist and key classification tasks. Hamel et al. [22] propose a deep learning architecture incorporating multiple feature aggregation functions for tag classification.

The approach proposed in this work resembles methods applying statistical models to quantised feature sequences [28, 30–32]. In contrast, we propose to compute summary statistics directly from estimated sequential models. Since the obtained statistics may be compared using a metric, our approach has the potential to be combined with indexing and hashing schemes for computationally efficient retrieval [37–39], while retaining information on temporal structure. Our method of computing multiple representations using downsampling resembles the approach proposed by Dieleman and Schrauwen [34].

3 Approach

Assume that we are given the audio feature vector sequence $\mathbf{V} = (\mathbf{v}_1, \dots, \mathbf{v}_T)$. Similar to the descriptor proposed in [40], as a means of quantifying the sequential complexity of \mathbf{V} , we compute the compression rate $R_\lambda(\mathbf{V})$,

$$R_\lambda(\mathbf{V}) = \frac{C(\mathbf{V}, \lambda)}{T} \quad (2)$$

where $C(\mathbf{V}, \lambda)$ denotes the number of bits required to represent \mathbf{V} , given a quantisation scheme with λ levels and using a specified string compression scheme. To obtain a length-invariant measure of sequential complexity, we normalise with respect to the sequence length T .

Given the i th track in our collection, we compute compression rates for feature sequences extracted from musical audio. We refer to the set of compression rates as *feature complexity descriptors* (FCDs). For features based on constant frame rate, we compute FCDs using the original feature sequence, in addition to FCDs computed on downsampled versions of the original sequence; we consider downsampling factors 1, 2, 4, 8. We distinguish among temporal resolutions using the labels FCD1, FCD2, FCD4, FCD8, respectively. For features based on variable frame rate, we compute FCDs with no further downsampling applied.

3.1 Similarity rating prediction

For the task of similarity rating prediction, assume that we have a distance metric which we use to compare descriptor vectors computed on pairs of tracks. We hypothesise that the pairwise distance between descriptors correlates with the similarity rating associated with track pairs. To predict similarity ratings we correspondingly take as our feature space pairwise distances between descriptor vectors. Henceforth, we use $\mathbf{r}_{i,n}$ to denote the n th descriptor vector computed for the i th track in our collection, with $1 \leq n \leq N$ and given a set of N available descriptor vectors. We compute separate descriptor vectors across audio features and across FCD resolutions, with each vector component corresponding to a quantisation granularity λ . As our predictive model, we apply the KNN algorithm to pairwise distances between descriptor vectors. We denote with $d_{\langle i,j \rangle, n}$ the distance between $\mathbf{r}_{i,n}$, $\mathbf{r}_{j,n}$, as computed using our assumed distance measure. Given a pair of tracks $\langle i, j \rangle$ whose similarity rating we seek to predict, we determine a set of nearest neighbours using the distance function $D(\langle i, j \rangle, \langle k, l \rangle)$ between pairs of tracks $\langle i, j \rangle$, $\langle k, l \rangle$,

$$D(\langle i, j \rangle, \langle k, l \rangle) = \left\{ \sum_{n=1}^N (\gamma_n |d_{\langle i,j \rangle, n} - d_{\langle k,l \rangle, n}|)^2 \right\}^{\frac{1}{2}} \quad (3)$$

where γ_n denotes a weighting coefficient for distances based on the n th descriptor. To predict similarity ratings, we compute the mode of discrete-valued ratings assigned to nearest neighbours, where we resolve ties by selecting the

rating with the lower value. In addition to feature weighting, we apply feature selection as a means of discarding redundant descriptors. We describe our feature weighting and selection approach in Section 4.1.3.

3.2 Song year prediction

For the task of song year prediction, we hypothesise that descriptor values correlate with the chart entry date of tracks. Following [41] we apply a linear regression model. Given the i th track in our collection, we predict the associated chart entry date y_i using a linear combination of components in descriptor vectors $\mathbf{r}_{i,n}$,

$$\hat{y}_i = \sum_{n=1}^N \beta_n^T \mathbf{r}_{i,n} + \alpha \quad (4)$$

where β_n denotes prediction coefficients for the n th descriptor vector as specified for similarity rating prediction, and where α denotes the model intercept. We motivate use of both proposed KNN and linear regression techniques as a means of evaluating the utility of FCDs for similarity based on a metric space.

4 Evaluation

For our evaluations, we use a collection of 15 473 entries from the American *Billboard Hot 100* singles popularity chart¹. Each entry in the dataset is represented by a track excerpt of approximately 30s of audio, and is annotated with a chart entry date. Chart entry dates span the years 1957–2010 ($M = 1982.9\text{y}$, $SD = 15.4\text{y}$).

For each track excerpt in the dataset, we extract a set of 25 audio features, using MIRToolbox [42] version 1.3.2 and using the framewise chromagram representation proposed by Ellis and Poliner [43]. With the exception of rhythmic features, which are computed using predicted onsets, features are based on a constant frame rate of 40Hz. Table 1 summarises the set of evaluated audio features.

In addition to FCDs, for each track excerpt we compute the mean and standard deviation, based on frame-level representation with no downsampling applied. We refer to the latter non-sequential descriptors as feature moment descriptors (FMDs). We compute FCDs as described in Section 3, where for the case of the vector-valued features chroma, MFCCs and delta-MFCCs we apply principal component analysis in track-wise fashion as a preliminary decorrelation step. We then quantise and compress each resulting component separately, before averaging obtained compression lengths across components. We quantise features by applying equal-frequency binning with $\lambda \in \{3, 4, 5\}$ levels.

We choose equal-frequency binning to ensure that obtained strings have a consistent stationary distribution; the obtained compression rates correspondingly are a function of temporal structure alone. The value $\log \lambda$ may be in-

¹<http://www.billboard.com>

terpreted as the theoretical compression rate for a temporally uncorrelated sequence. We compress symbol sequences using the *prediction by partial match* (PPM) algorithm², described in [50], where we set the model order to 5 symbols. We motivate the stated parameter choices for model order and λ based on preliminary evaluations.

4.1 Similarity rating prediction

We evaluate similarity rating prediction using annotations collected for a subset of the chart music dataset. Prior to our investigations, we obtained a total of 7784 pairwise similarity ratings from 456 subjects participating in a web-based listening test³. Subjects were asked to quantify pairwise musical similarity between pairs of track excerpts using a five-point ordinal scale, with rating ‘1’ corresponding to ‘not similar’ and rating ‘5’ corresponding to ‘very similar’. We assume that subjects have a pre-existing similarity scale which they use to perform ratings. Correspondingly, we omit any training step from the rating process. Fig. 1 displays a histogram of similarity ratings, where for each rating count we estimate the standard error using bootstrap sampling [51]; in this work, we base bootstrap estimates on 10^4 samples.

When presenting track pairs to listeners, we select the first song in each pair using uniform sampling. For the second song in each pair, we again apply uniform sampling, however we bias towards proximate chart entry times by restricting the permissible chart entry deviation to $\leq 1y$ with probability 0.9. We bias as a means of controlling for historical changes in audio production, which might affect similarity ratings [52]. We obtain a median of 6 ratings per subject, with each rating corresponding to a unique track pair.

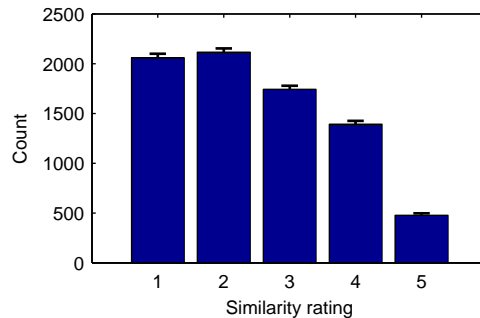


Figure 1: Histogram of similarity ratings. Error bars indicate standard errors estimated using bootstrap sampling.

To assess the consistency of similarity ratings, we collected an additional set of similarity ratings under controlled experimental conditions, involving 12

²<http://www.cs.technion.ac.il/~ronbeg/vmm/index.html>

³<http://webprojects.eecs.qmul.ac.uk/matthiasm/audioquality-pre/check.php>

subjects aged 21y–42y. Subjects were assessed using the Ollen musical sophistication index (OMSI) [53]. We obtain a median OMSI score of 241, with an associated median of 0.75 years of formal musical training. To avoid subject fatigue, we imposed no minimum number of ratings per subject, and collected ratings during two 30-minute sessions. We selected stimuli by sampling uniformly from the set of track pairs for which we have prior ratings. Across subjects, we obtain a median of 42 ratings ($M = 45.4$, $SD = 29.3$). Thus, we obtain a total of 509 additional similarity ratings, corresponding to 6.5% coverage of web-based similarity ratings.

We quantify the agreement between controlled-condition and web-sourced similarity ratings using Kendall’s correlation coefficient τ_b , as defined in (5). Based on the box plot shown in Fig. 2, exploratory analysis of subject-wise τ_b reveals a median correlation of 0.284. To obtain a measure of rating consistency among web-sourced ratings, we aggregate controlled-condition ratings across subjects and correlate with web-sourced ratings. We obtain a correlation of 0.273, with $p < 0.001$ based on a permutation test for the hypothesis of no correlation. We then compute a confidence interval for the obtained sample correlation by applying bootstrap sampling. At the 95% level, we obtain correlations in the range $[0.205, 0.337]$. Subsequently, we consider the correlation 0.337 an upper bound on attainable accuracy using our proposed method of similarity rating prediction. As a second measure of rating agreement, we compute Spearman’s correlation coefficient ρ_s , where we obtain $\rho_s = 0.328$ for ratings aggregated across subjects. Analogously by applying bootstrap sampling, at the 95% level we obtain correlations in the range $[0.247, 0.404]$. We consider the correlation 0.404 a bound on attainable accuracy based on ρ_s .

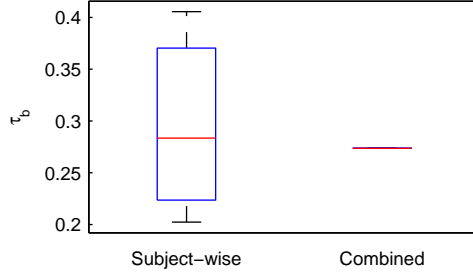


Figure 2: *Left*: Box plot of subject-wise Kendall τ_b between controlled-condition similarity ratings and web-sourced similarity ratings. Central mark represents median subject-wise correlation. Box limits correspond to 25th and 75th percentiles. Whiskers extend to 1.5 times interquartile range. *Right*: Correlation between aggregated controlled-condition similarity ratings and web-sourced similarity ratings.

4.1.1 Distance measures

We predict similarity ratings by applying KNN to pairwise Euclidean distances between descriptor vectors, using the approach described in Section 3.1. As an additional baseline distance measure, using (1) and assuming Gaussianity and diagonal covariance, we compute the KLD on pairs of FMDs.

As a baseline distance accounting for temporal structure, we compute the cross-prediction error between audio feature sequences, with each feature sequence represented at the original frame level. Following [54], we apply state space embedding [55] separately to pairs of feature sequences. Given feature vectors $(\mathbf{v}_1, \dots, \mathbf{v}_T)$ each with dimensionality h , state space embedding produces higher-dimensional feature vectors with dimensionality dh by stacking d consecutive vectors $\mathbf{v}_{t-d}, \dots, \mathbf{v}_{t-1}$ at each time step t . We perform cross-predictions by determining sequential successors of nearest neighbours in the embedded space, using the approach given in [56]. As a distance measure between predicted and observed feature sequences, we compute the normalised mean square error [54]. We consider parameter $d \in \{8, 12, 16, 20\}$ and report results for $d = 12$, which yields highest average correlation between computed pairwise distances and similarity annotations.

4.1.2 Performance statistics

To quantify the accuracy of similarity rating prediction, we compute Kendall's τ_b and Spearman's ρ_s , both which express the amount of correlation between predicted and annotated similarity ratings. We define Kendall's τ_b as follows. Assume that we have sequences $\mathcal{Q} = (q_1, \dots, q_M)$, $\mathcal{O} = (o_1, \dots, o_M)$. The pair $d_{i,j} = ((q_i, o_i), (q_j, o_j))$ is termed *concordant*, if $q_i > q_j$ and $o_i > o_j$, or if $q_i < q_j$ and $o_i < o_j$. Analogously, $d_{i,j}$ is termed *discordant*, if $q_i < q_j$ and $o_i > o_j$, or if $q_i > q_j$ and $o_i < o_j$. Kendall's τ_b is defined as

$$\tau_b = \frac{M_c - M_d}{\sqrt{(M_p - M_q)(M_p - M_o)}} \quad (5)$$

where M_c , M_d respectively denote the number of concordant and discordant pairs and where $M_p = \frac{1}{2}M(M-1)$ denotes the total number of pairs. Terms M_q , M_o respectively denote the number of pairs with tied (q_i, q_j) and with tied (o_i, o_j) . In the denominator, the normalisation is with respect to the geometric mean of adjusted pair counts $(M_p - M_q)$, $(M_p - M_o)$. Yielding values in the range $[-1, 1]$, τ_b may be interpreted as an estimate of the difference in probability of sampling a concordant pair versus sampling a discordant pair, given the set of concordant and discordant pairs contained in $(\mathcal{Q}, \mathcal{O})$.

As a second measure of prediction accuracy, we compute Spearman's ρ_s , corresponding to the product-moment correlation coefficient between separately ranked \mathcal{Q} , \mathcal{O} . We assign unique ranks to tied values, before computing average ranks across tied values. Note that in contrast to τ_b , the value of ρ_s is a function of assigned ranks. Correspondingly, in the presence of ties τ_b may be viewed as a more appropriate means of comparing ordinal sequences [57]. Nevertheless,

we compute ρ_s , since its square yields a direct interpretation as proportion of explained variance between assigned ranks.

4.1.3 Model estimation

We evaluate similarity rating prediction by applying hold-out validation to web-sourced annotations. We use 60% of annotations for training, with the remainder of annotations used for testing. In our training step, we standardise distances and compute weights γ_n in (3) using the correlation between distance and similarity rating; we consider both τ_b and ρ_s .

In our training step, we select descriptors using a modification of the *minimal-redundancy-maximal-relevance* (mRMR) criterion [58]. Assume that we are given an L -dimensional feature space, which we denote with $\mathcal{D} = \{\delta_1, \dots, \delta_L\}$, in addition to a response variable c . In our case, our L -dimensional feature space corresponds to pairwise distances between descriptors computed separately across audio features; our response variable corresponds to pairwise similarity annotations. According to mRMR we seek a subset $S \subseteq \mathcal{D}$ which maximises the difference function $\Phi(D, R) = D - R$ where D, R respectively correspond to *relevance* and *redundancy*,

$$D(S, c) = \frac{1}{|S|} \sum_{\delta_i \in S} I(\delta_i; c) \quad (6)$$

$$R(S) = \frac{1}{|S|^2} \sum_{\delta_i, \delta_j \in S} I(\delta_i; \delta_j). \quad (7)$$

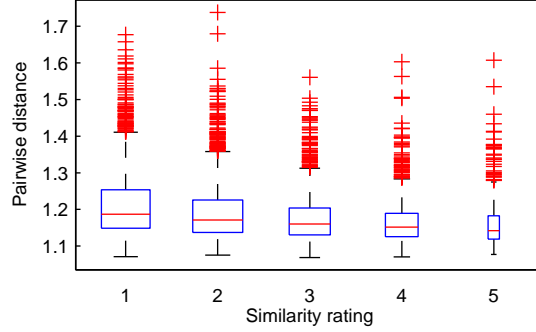
In (6), $I(\delta_i; c)$ denotes mutual information between distances computed on descriptors δ_i , and similarity annotations c . In our approach, we assume that for informative descriptors, the rated similarity is ideally a monotonic function of distance between descriptors. Correspondingly, in place of mutual information we instead employ our chosen measures of correlation τ_b, ρ_s . Using the incremental mRMR search procedure given in [58], we obtain L solution candidates $S_1 \subset \dots \subset S_\ell \subset \dots \subset S_L$. We then select S_ℓ so as to maximise the KNN prediction accuracy within our training data, again using hold-out validation. We favour compact descriptor subsets by minimising $|S_\ell|$ among $P = 5$ best solutions. Following [59], to further promote compactness we apply backward sequential feature selection as a final step.

We apply feature selection separately to sets of distances between descriptor vectors, as specified in Table 2. Note that we compute FCD vectors separately across temporal resolutions and across audio features. Based on a set of 25 audio features, given a pair of tracks we thus obtain a total of 100 distances between compression-based descriptor vectors. Furthermore, note that when combining sets of descriptors we aggregate among obtained distances. Thus given a pair of tracks, when combining sets 1, 3, 4 as specified in Table 2, we obtain 150 distances. As given in (3), we weight the distances individually.

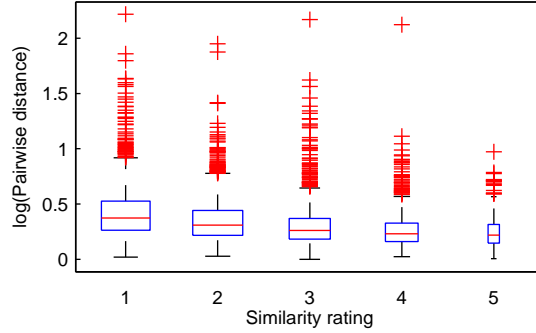
We report results for KNN using $K = 50$, based on optimising ρ_s with respect to the training partition and using Euclidean distances between FMDs to predict similarity ratings.

4.1.4 Results

Fig. 3 displays the result of exploratory analysis, in which we plot pairwise distances averaged across features, against pairwise similarity ratings. We consider FCDs computed without downsampling and FMDs, respectively compared using Euclidean distance and standardised KLD. For both descriptors, we observe a monotonically decreasing trend in median, 25th and 75th percentile ranges against increasing similarity rating.



(a) FCDs (FCD1) compared using Euclidean distance



(b) FMDs compared using KLD

Figure 3: Box plot of pairwise distances against web-sourced pairwise similarity ratings, obtained using (a) FCDs computed without downsampling and (b) FMDs. Distances averaged across features. Crosses represent outliers. Box widths proportional to number of observations.

We next examine the correlation between descriptor distances and similarity ratings for individual features. Fig. 4 depicts rank correlations ρ_s across features for FCDs and FMDs, where we compare FMDs using both Euclidean distance

Chroma (Ellis and Poliner)	-0.04*	0.04*	0.03*	0.17*	0.18*	0.15*	0.08*
dynamics.rms	-0.09*	0.08*	0.09*	0.11*	0.10*	0.09*	0.07*
rhythm.tempo	-0.07*	0.04*	0.02	0.01	0.01	0.01	0.01
rhythm.attack.time	-0.05*	0.05*	0.03*	0.03	0.03	0.03	0.03
rhythm.attack.slope	-0.03	0.03*	0.01	0.05*	0.05*	0.05*	0.05*
spectral.centroid	-0.12*	0.10*	0.05*	0.07*	0.06*	0.06*	0.03*
spectral.brightness	-0.12*	0.13*	0.09*	0.09*	0.08*	0.05*	0.02
spectral.spread	-0.11*	0.10*	0.08*	0.07*	0.10*	0.07*	0.04*
spectral.skewness	-0.04*	0.06*	0.07*	0.11*	0.10*	0.08*	0.03*
spectral.kurtosis	-0.03	0.06*	0.06*	0.10*	0.10*	0.07*	0.05*
spectral.rolloff95	-0.08*	0.06*	0.04*	0.08*	0.07*	0.05*	0.05*
spectral.rolloff85	-0.11*	0.09*	0.05*	0.08*	0.07*	0.04*	0.03
spectral.spectentropy	-0.13*	0.12*	0.09*	0.07*	0.08*	0.06*	0.03*
spectral.flatness	-0.09*	0.08*	0.04*	0.07*	0.06*	0.05*	0.04*
spectral.roughness	-0.03	0.04*	0.06*	0.09*	0.10*	0.07*	0.04*
spectral.irregularity	-0.05*	0.06*	0.07*	0.10*	0.11*	0.07*	0.03*
spectral.mfcc	-0.14*	0.15*	0.07*	0.18*	0.19*	0.15*	0.08*
spectral.dmfcc	-0.14*	0.16*	0.03	0.08*	0.04*	0.06*	0.05*
spectral.ddmfcc	-0.14*	0.16*	0.03	0.04*	0.01	0.04*	0.04*
timbre.zerocross	-0.12*	0.11*	0.07*	0.05*	0.06*	0.04*	0.02
timbre.spectralflux	-0.19*	0.18*	0.04*	0.09*	0.08*	0.04*	0.04*
tonal.chromagram.centroid	-0.10*	0.10*	0.12*	0.07*	0.07*	0.07*	0.03*
tonal.keyclarity	-0.15*	0.15*	0.13*	0.14*	0.11*	0.07*	0.01
tonal.mode	-0.08*	0.10*	0.09*	0.15*	0.13*	0.07*	0.01
tonal.hcdf	-0.11*	0.12*	0.03*	0.09*	0.05*	0.03	0.02
	FMDs (Euclidean)	FMDs (KLD)	Frame-level cross-prediction	FCD1	FCD2	FCD4	FCD8

Figure 4: Feature-wise absolute correlation $|\tau_b|$ between pairwise distances and web-sourced similarity annotations. Pairwise distances respectively obtained using FMDs compared using Euclidean distance and KLD (first and second columns), cross-prediction (third column), Euclidean distance applied to FCDs (remaining columns). Starred entries indicate significance, where we apply Bonferroni correction to $\alpha = 0.05$.

and KLD. In addition to FMDs, as described in Section 4.1.1 we consider as a baseline the cross-prediction error.

We observe that FCDs and FMDs both yield maximum correlation 0.19 (comparing FCD2 to FMDs, with both distances computed using Euclidean distance); similarly, FMDs compared using KLD yield maximum correlation 0.18. Across descriptors, with $\alpha = 0.05$ and applying Bonferroni correction, the majority of features yield significant correlations. In contrast, for cross-prediction, effect sizes are comparatively small. Comparing descriptors, for FCD2 we observe correlations exceeding 0.1 for 9 features, and for 12 features for the case of FMDs compared either using KLD or Euclidean distance. On average, FMDs yield greater correlation compared to FCD1 (0.095 versus 0.088). However, for specific features FCDs yield higher correlation than FMDs. Comparing FCDs amongst temporal resolutions, we observe a monotonically decreasing relation-

ship between downsampling factor and average correlation.

Fig. 5 displays a comparison of similarity rating prediction accuracy, where for each descriptor set we apply feature selection as described in Section 4.1.3. We perform feature selection and weighting with respect to τ_b and alternatively ρ_s . In particular, we consider the performance gain incurred by including FCDs in our model.

We observe that FCDs are outperformed by FMDs compared using either KLD or Euclidean distance alone. However, a combination of FCDs and FMDs outperforms evaluated combinations employing FMDs alone. By incorporating compression descriptors, compared to FMDs based on aggregated KLD and Euclidean distance, we obtain absolute gains in correlation of 0.049, 0.038, with respect to ρ_s , τ_b . The corresponding relative performance gains are 18.5%, 16.5%. As suggested by Fig. 4, we observe that cross-prediction yields comparatively low prediction performance. Qualitative performance differences between descriptor sets hold for both τ_b , ρ_s . We test for differences between correlations by applying bootstrap sampling to predicted and observed similarity ratings, from which in turn we estimate standard errors of ρ_s , τ_b . Based on a one-way analysis of variance with Tukey-Kramer post-hoc analysis and setting $\alpha = 0.05$, we reject the hypothesis of no difference between correlations across all considered pairs, for both ρ_s , τ_b .

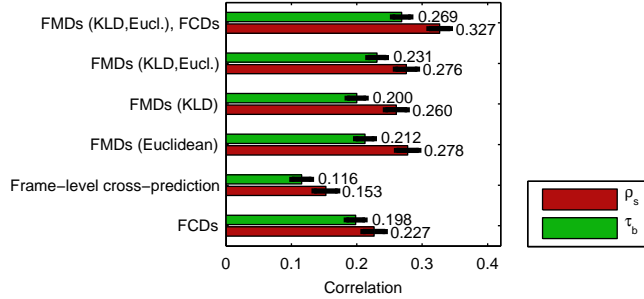


Figure 5: Similarity rating prediction accuracy using combined descriptors. Error bars denote standard errors, obtained by bootstrap sampling pairs of predicted and observed similarity ratings and computing ρ_s , τ_b on each sample.

Fig. 6 displays a matrix of selected descriptors across features and descriptor classes, where we consider the best-performing model evaluated in the previous Fig. 5, based on ρ_s . For each selected descriptor, we quantify utility in terms of correlation gain obtained by including the descriptor, with respect to the evaluation data contained in the training partition. Comparing FMDs and FCDs, we observe that both FCDs and FMDs are selected within individual features. FCDs appear to be selected across diverse temporal resolutions, with emphasis on higher temporal resolutions. We observe that multiple FCD resolutions are selected within the same feature.

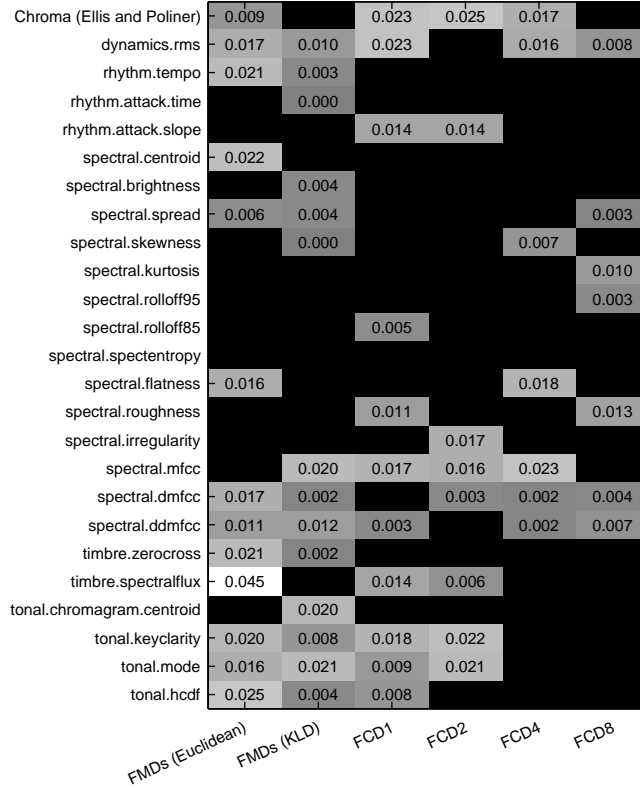


Figure 6: Binary indicator matrix of selected descriptors for similarity rating prediction, based on ρ_s . Candidate descriptor set comprised of FCDs compared using Euclidean distance, and FMDs compared using Euclidean distance and KLD. Dark markers indicate absence of descriptors in final model. See main text for description of numerical entries.

4.2 Song year prediction

For song year prediction, we compute FCDs and FMDs as performed for similarity rating prediction. We use chart entry dates as our annotation data and apply the linear regression model given in (4). Fig. 1 displays a histogram of chart entry dates.

4.2.1 Model estimation

To evaluate our descriptors for song year prediction, we partition the dataset into random training and testing subsets, where we ensure that title or artist strings are not duplicated across subsets. We apply the aforementioned filtering procedure to control for potential cover version and album effects, in addition to any analogous effects at the level of artists [60]. The resulting training and

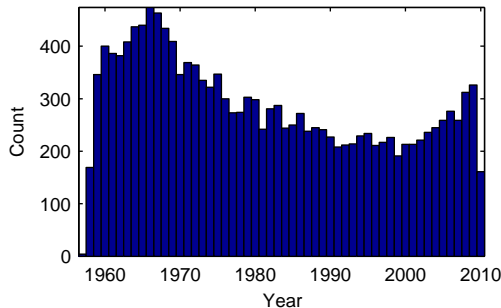


Figure 7: Histogram of chart entry dates.

testing datasets consist of 10 728 and 4 745 tracks respectively.

As was performed for similarity rating prediction, we standardise descriptors with respect to training data. We estimate regression coefficients β_n given in (4) using L2 regularisation [61]. We optimise the associated regularisation parameter with respect to the mean square error, by applying 10-fold cross validation to the training data. We deem as outliers descriptor values in the training data exceeding 10 standard deviations beyond the 99th percentile. We replace such outliers with imputed values, using the KNN algorithm. Furthermore, we threshold predictions to fall within the range [1957y .. 2010y]

We apply regression to sets of descriptor vectors, as specified in Table 3. As performed for similarity rating prediction, we compute FCDs separately across temporal resolutions and across audio features. In contrast, we apply regression directly to descriptor vectors without the intermediate step of computing distances. Based on a set of 25 audio features, given a single track we obtain a total of 300 scalar-valued FCDs, for each of which we estimate a single regression coefficient. Note that since we represent FMDs using the mean and standard deviation, we estimate two regression coefficients for each univariate audio feature. For FMDs, it follows that we estimate 24 regression coefficients for MFCCs and chroma features.

In addition to the year prediction task based on individual tracks, we evaluate prediction performance when considering groups of tracks. We select groups of tracks by applying a non-overlapping sliding window to chart entry dates. We then take as descriptor vector $\mathbf{r}'_{w,n}$ the average

$$\mathbf{r}'_{w,n} = \frac{1}{|C_w|} \sum_{i \in C_w} \mathbf{r}_{i,n} \quad (8)$$

where C_w denotes the set of tracks at window position w . We apply the windowing procedure separately to training and testing data sets. For a given window size, we proceed as described in Section 4.2.1; given the obtained regression model and given descriptor vectors at window position z in the testing data, we seek to predict the associated window centre y'_z .

4.2.2 Performance statistics

We quantify prediction accuracy with respect to annotated chart entry dates, using the mean absolute error (MAE) and root mean square error (RMSE) statistics.

4.2.3 Results

Fig. 8 displays the result of exploratory analysis for song year prediction, where for FMDs and FCDs we group descriptor values across time, by applying a non-overlapping 2-year sliding window to chart entry dates. We restrict analysis to obtained spectral spread features [42]. The resulting year-wise box plots suggest that the examined descriptors are non-stationary with respect to chart entry dates, exhibiting distinct trends. To examine the behaviour of descriptors at a finer time scale, we apply a non-overlapping 30-day sliding window to chart entry dates, where at each window position we compute the mean descriptor value. Examining the sample autocorrelation of the resulting time series for lags in the range $[1 \dots 15]$, we observe weaker correlations for FCDs compared to FMDs. Yet, both autocorrelations exhibit slowly decaying autocorrelations (Fig. 9), characteristic of a non-stationary time series [62]. Following the method of Box and Jenkins [63], we attempt to attain stationarity by applying first-order differencing to the time series. However, we observe autocorrelation close to -0.5 at unit lag, suggesting that the time series have been overdifferenced [62]. We interpret these observations as evidence for a non-trivial, trend-exhibiting process governing observed descriptor values [64].

Table 4 summarises the accuracy of song year prediction using MAE and RMSE statistics. Quantified using either MAE or RMSE, song year prediction based on FMDs outperforms prediction using FCDs alone. However, we observe that a combination of FMDs and FCDs yields the highest prediction accuracy. By incorporating FCDs we observe performance gains of 9.9%, 8.6% relative to FMDs, in terms of MAE and RMSE. As performed in Section 4.1.4, we test for differences among prediction accuracies by applying bootstrap sampling to predicted and observed chart entry times, from which we estimate standard errors of MAE and RMSE statistics. Again using one-way analysis of variance with Tukey-Kramer post-hoc analysis and setting $\alpha = 0.05$, we reject the hypothesis of no difference between prediction accuracies across all pairs, for both MAE and RMSE.

Fig. 10 displays regression coefficients obtained using unwindowed chart entry dates. We scale coefficients by descriptor standard deviations, before computing magnitudes and normalising to sum to one. Thus computed, we interpret coefficients in terms of predictive utility across individual audio features. In addition, we consider the utility of FCDs across time scales, compared to FMDs. Summed across features, we observe that compared to FCD1, FMDs are weighted more strongly (0.538 versus 0.218). Further examining relative weightings, we observe a prevalence of weight assigned to FCD1 compared to higher downsampling factors. However, we observe that individual features

may be weighted relatively strongly across multiple temporal scales. Note from Table 2 that for chroma features, MFCCs and derivatives, FMD weights are summed across 24 prediction coefficients, compared to 3 coefficients for FCDs.

In Fig. 11 we examine prediction accuracy in response to windowed descriptors, as described in Section 4.1.3 and quantified using MAE. For increasing window size, performance improves monotonically across all considered descriptor sets. Across considered window sizes, using combined FCDs and FMDs we observe a mean performance gain of 15.1%, relative to using FMDs alone.

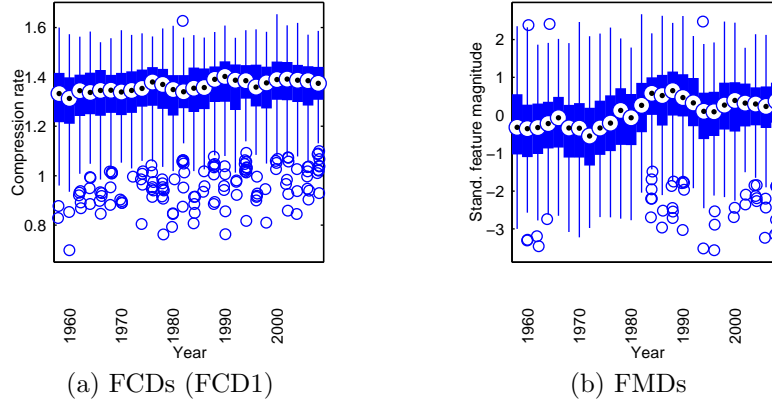


Figure 8: Box plots of FCDs and FMDs computed using spectral spread features, with FCDs computed without downsampling. Each box corresponds to the position of a non-overlapping 1-year window applied to chart entry dates.

5 Conclusions

We have considered the problem of determining musical similarity, using feature sequences extracted from musical audio. In particular, we have considered musical similarity in the context of two low-specificity content retrieval tasks, namely similarity rating prediction and song year prediction. To this end, we have evaluated the utility of sequential complexity as a descriptor for quantifying musical similarity.

For both considered tasks, we observe that sequential complexity descriptors predict the outcome variable. Furthermore, in combination with feature moment descriptors, sequential complexity descriptors improve prediction accuracy with respect to the baseline. The results confirm that our proposed descriptors capture musically relevant information and that temporal structure is relevant in our chosen domain. Consequently, our results show that sequential complexity may be used to improve the accuracy of low-specificity content retrieval based on bag-of features approaches.

Our proposed descriptors are computed in an unsupervised manner and may be implemented efficiently, requiring $O(n)$ time complexity for each track [65].

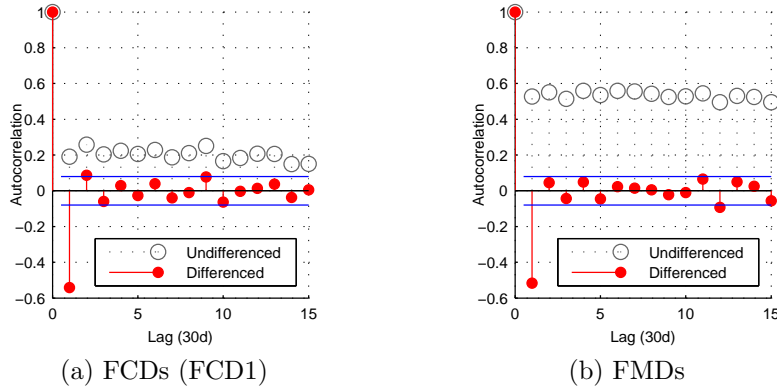


Figure 9: Sample autocorrelation of undifferenced and differenced FCD, FMD averages. Descriptor averages obtained by applying non-overlapping 30-day window to chart entry dates. Descriptors computed on spectral spread features, with FCDs computed without downsampling. Horizontal bars indicate 95% confidence intervals under the assumption of Gaussian white noise for differenced time series.

In addition, our proposed descriptors have similar dimensionality compared to feature moment descriptors. Since our descriptors may be computed off-line or incrementally and thereafter combined with indexing methods as proposed in [37–39], we deem them potentially applicable in large-scale content retrieval systems.

Similar to results obtained in [22,33,34,66], our results using sequential complexity descriptors suggest that an approach based on multiple temporal resolutions is advantageous for determining musical similarity. As an alternative to downsampled features, we initially employed beat-synchronous representations, which yielded comparatively small gains in prediction accuracy, when combined with original frame-based features. This result suggests that for our chosen domain, temporal structure at short time scales is more advantageous, compared to temporal structure at the metrical level. For future work, we aim to examine in closer detail the utility of representing features at multiple time scales.

For similarity rating prediction, note that by biasing towards tracks with proximate chart entry dates, we attempt to control for historical changes in audio production. For song year prediction, where we do not control in the described manner, audio production may confound the association between musical similarity and chart entry date. We acknowledge that in both cases, audio production may confound the association between similarity measures and respective outcome variables, as observed in [52]. For future work, we aim to measure the degree of confounding by introducing suitable audio degradations [67].

Finally, the present work considers only a single sequential complexity measure, estimated using a single algorithm. While we obtained similar results

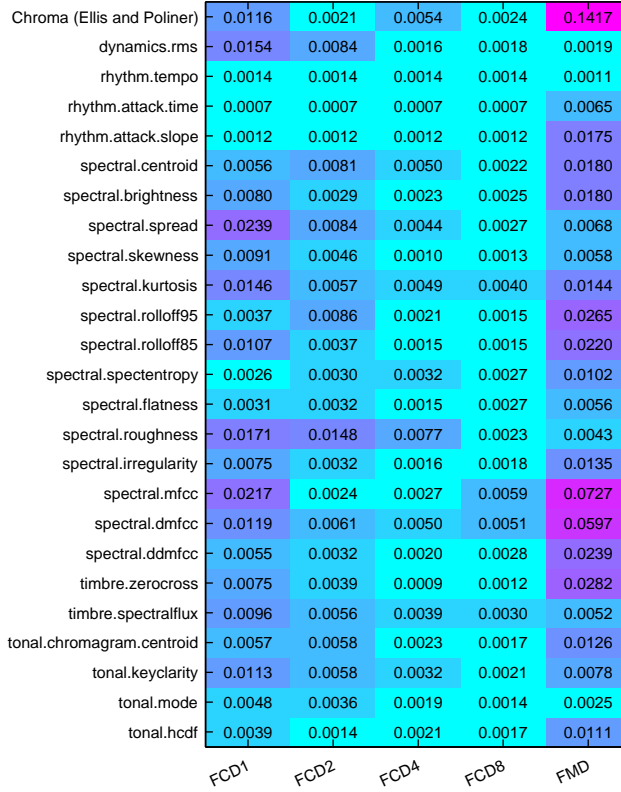


Figure 10: Normalised regression coefficient magnitudes, estimated using L2 regularisation, for task of song year prediction. Coefficient magnitudes scaled with respect to standard deviation of training data before normalising.

using the Lempel-Ziv algorithm in initial evaluations, it is conceivable that using multiple compression algorithms may reduce the error variance of estimated sequential complexity. Using alternative classification tasks, we aim to evaluate whether multiple compressors yield an improvement in prediction accuracy. In addition, we aim to evaluate alternative measures of sequential complexity [68–70].

6 Acknowledgements

This work benefited from advice and comments from Andrew J. R. Simpson, Dan Stowell, Anssi Klapuri, Mark D. Plumbley, and Armand Leroi.

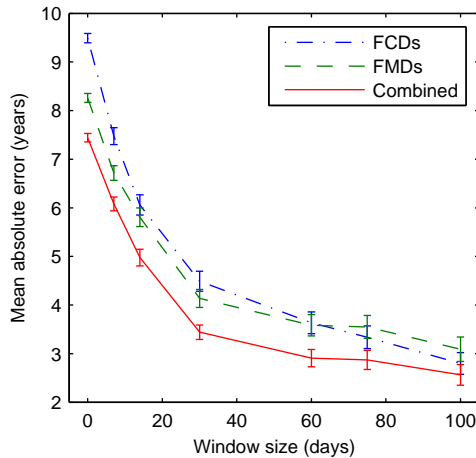


Figure 11: Song year prediction accuracy obtained using windowed descriptors, in response to window size. Error bars denote standard errors.

References

- [1] M. Casey, R. Veltkamp, M. Goto, M. Leman, C. Rhodes, and M. Slaney, “Content-based music information retrieval: Current directions and future challenges,” *Proc. IEEE*, vol. 96, no. 4, pp. 668–696, 2008.
- [2] Z. Fu, G. Lu, K. Ting, and D. Zhang, “A survey of audio-based music classification and annotation,” *IEEE Trans. Multimedia*, vol. 13, no. 2, pp. 303–319, 2011.
- [3] O. Celma, “Music recommendation and discovery in the long tail,” Ph.D. dissertation, Universitat Pompeu Fabra, Barcelona, Spain, 2009.
- [4] P. Cano, E. Batlle, T. Kalker, and J. Haitsma, “A review of audio fingerprinting,” *Journal of VLSI Signal Processing Systems for Signal, Image and Video Technology*, vol. 41, no. 3, pp. 271–284, 2005.
- [5] J. Serrà, “Identification of versions of the same musical composition by processing audio descriptions,” Ph.D. dissertation, Universitat Pompeu Fabra, Barcelona, Spain, 2011.
- [6] N. Scaringella, G. Zoia, and D. Mlynek, “Automatic genre classification of music content: a survey,” *IEEE Signal Processing Magazine*, vol. 23, no. 2, pp. 133–141, 2006.
- [7] Y. Kim, E. Schmidt, R. Migneco, B. Morton, P. Richardson, J. Scott, J. Speck, and D. Turnbull, “Music emotion recognition: A state of the art review,” in *Proc. 11th Intern. Society for Music Information Retrieval Conf. (ISMIR)*, 2010, pp. 255–266.

- [8] J. Aucouturier, B. Defreville, and F. Pachet, “The bag-of-frames approach to audio pattern recognition: A sufficient model for urban soundscapes but not for polyphonic music,” *The Journal of the Acoustical Society of America*, vol. 122, p. 881, 2007.
- [9] G. Tzanetakis and P. Cook, “Musical genre classification of audio signals,” *IEEE Trans. Speech and Audio Processing*, vol. 10, no. 5, pp. 293–302, 2002.
- [10] T. Li and M. Ogihara, “Toward intelligent music information retrieval,” *IEEE Trans. Multimedia*, vol. 8, no. 3, pp. 564–574, 2006.
- [11] K. West, S. Cox, and P. Lamere, “Incorporating machine-learning into music similarity estimation,” in *Proc. 1st ACM workshop on Audio and Music Computing Multimedia*, 2006, pp. 89–96.
- [12] M. Slaney, K. Weinberger, and W. White, “Learning a metric for music similarity,” in *Proc. 9th Intern. Conf. Music Information Retrieval (ISMIR)*, 2008, pp. 313–318.
- [13] B. Logan and A. Salomon, “A music similarity function based on signal analysis,” in *Proc. Intern. Conf. Multimedia and Expo. (ICME)*, 2001.
- [14] J. Aucouturier and F. Pachet, “Music similarity measures: What’s the use?” in *Proc. 3rd Intern. Conf. Music Information Retrieval (ISMIR)*, 2002.
- [15] A. Berenzweig, B. Logan, D. Ellis, and B. Whitman, “A large-scale evaluation of acoustic and subjective music-similarity measures,” *Computer Music Journal*, vol. 28, no. 2, pp. 63–76, 2004.
- [16] M. Mandel and D. Ellis, “Song-level features and support vector machines for music classification,” in *Proc. 6th Intern. Conf. Music Information Retrieval (ISMIR)*, 2005, pp. 594–599.
- [17] F. Vignoli and S. Pauws, “A music retrieval system based on user driven similarity and its evaluation,” in *Proc. 6th Intern. Conf. Music Information Retrieval (ISMIR)*, 2005, pp. 272–279.
- [18] M. Levy and M. Sandler, “Lightweight measures for timbral similarity of musical audio,” in *Proc. 1st ACM workshop on Audio and music computing multimedia*, 2006, pp. 27–36.
- [19] S. McAdams, S. Winsberg, S. Donnadiou, G. De Soete, and J. Krimphoff, “Perceptual scaling of synthesized musical timbres: Common dimensions, specificities, and latent subject classes,” *Psychological Research*, vol. 58, no. 3, pp. 177–192, 1995.
- [20] K. Seyerlehner, G. Widmer, and T. Pohle, “Fusing block-level features for music similarity estimation,” in *Proc. of the 13th Int. Conf. on Digital Audio Effects (DAFx-10)*, 2010, pp. 225–232.

- [21] F. Mörchén, A. Ultsch, M. Thies, and I. Lohken, “Modeling timbre distance with temporal statistics from polyphonic music,” *IEEE Trans. Audio, Speech and Language Processing*, vol. 14, no. 1, pp. 81–90, 2006.
- [22] P. Hamel, S. Lemieux, Y. Bengio, and D. Eck, “Temporal pooling and multiscale learning for automatic annotation and ranking of music audio.” in *Proc. 12th Intern. Society for Music Information Retrieval Conf. (ISMIR)*, 2011, pp. 729–734.
- [23] J. Wülfing and M. Riedmiller, “Unsupervised learning of local features for music classification.” in *Proc. 13th Intern. Society for Music Information Retrieval Conf. (ISMIR)*, 2012, pp. 139–144.
- [24] E. Pampalk, “Computational models of music similarity and their application in music information retrieval,” Ph.D. dissertation, Vienna University of Technology, Vienna, Austria, 2006.
- [25] C. Lee, J. Shih, K. Yu, and H. Lin, “Automatic music genre classification based on modulation spectral analysis of spectral and cepstral features,” *IEEE Trans. Multimedia*, vol. 11, no. 4, pp. 670–682, 2009.
- [26] A. Meng, P. Ahrendt, J. Larsen, and L. Hansen, “Temporal feature integration for music genre classification,” *IEEE Trans. Audio, Speech and Language Processing*, vol. 15, no. 5, pp. 1654–1664, 2007.
- [27] E. Coviello, Y. Vaizman, A. Chan, and G. Lanckriet, “Multivariate autoregressive mixture models for music auto-tagging.” in *Proc. 13th Intern. Society for Music Information Retrieval Conf. (ISMIR)*, 2012, pp. 547–552.
- [28] M. Li and R. Sleep, “Genre classification via an LZ78-based string kernel.” in *Proc. 6th Intern. Conf. Music Information Retrieval (ISMIR)*, 2005, pp. 252–259.
- [29] J. Ziv and A. Lempel, “Compression of individual sequences via variable-rate coding,” *IEEE Trans. Information Theory*, vol. 24, no. 5, pp. 530–536, 1978.
- [30] J. Reed and C. Lee, “On the importance of modeling temporal information in music tag annotation,” in *Proc. IEEE Intern. Conf. Acoustics, Speech and Signal Process. (ICASSP)*. IEEE, 2009, pp. 1873–1876.
- [31] T. Langlois and G. Marques, “A music classification method based on timbral features.” in *Proc. 10th Intern. Society for Music Information Retrieval Conf. (ISMIR)*, 2009, pp. 81–86.
- [32] J. Ren and J. Jang, “Discovering time-constrained sequential patterns for music genre classification,” *IEEE Trans. Audio, Speech and Language Processing*, vol. 20, no. 4, pp. 1134–1144, 2012.

- [33] R. Foucard, S. Essid, M. Lagrange, and G. Richard, “Multi-scale temporal fusion by boosting for music classification,” in *Proc. 12th Intern. Society for Music Information Retrieval Conf. (ISMIR)*, 2011, pp. 663–668.
- [34] S. Dieleman and B. Schrauwen, “Multiscale approaches to music audio feature learning,” in *Proc. 14th Intern. Society for Music Information Retrieval Conf. (ISMIR)*, 2013, pp. 3–8.
- [35] M. Mauch and M. Levy, “Structural change on multiple time scales as a correlate of musical complexity,” in *Proc. 12th Intern. Society for Music Information Retrieval Conf. (ISMIR)*, 2011, pp. 489–494.
- [36] S. Dieleman, P. Brakel, and B. Schrauwen, “Audio-based music classification with a pretrained convolutional network,” in *Proc. 12th Intern. Society for Music Information Retrieval Conf. (ISMIR)*, 2011, pp. 669–674.
- [37] M. Slaney and M. Casey, “Locality-sensitive hashing for finding nearest neighbors,” *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 128–131, 2008.
- [38] C. Rhodes, T. Crawford, M. Casey, and M. d’Inverno, “Investigating music collections at different scales with AudioDB,” *Journal of New Music Research*, vol. 39, no. 4, pp. 337–348, 2010.
- [39] J. Schlüter, “Learning binary codes for efficient large-scale music similarity search,” in *Proc. 14th Intern. Society for Music Information Retrieval Conf. (ISMIR)*, 2013, pp. 581–586.
- [40] S. Streich, “Automatic characterization of music complexity: a multifaceted approach,” Ph.D. dissertation, Universitat Pompeu Fabra, Barcelona, Spain, 2006.
- [41] T. Bertin-Mahieux, D. Ellis, B. Whitman, and P. Lamere, “The million song dataset,” in *Proc. 12th Intern. Society for Music Information Retrieval Conf. (ISMIR)*, 2011, pp. 591–596.
- [42] O. Lartillot and P. Toivainen, “A Matlab toolbox for musical feature extraction from audio,” in *Proc. Intern. Conf. Digital Audio Effects (DAFx)*, 2007, pp. 237–244.
- [43] D. P. W. Ellis and G. Poliner, “Identifying ‘cover songs’ with chroma features and dynamic programming beat tracking,” in *Proc. IEEE Intern. Conf. Acoustics, Speech and Signal Process. (ICASSP)*, 2007, pp. 1429–1432.
- [44] R. Plomp and W. Levelt, “Tonal consonance and critical bandwidth,” *Journal of the Acoustical Society of America*, vol. 38, p. 548, 1965.
- [45] K. Jensen, “Timbre models of musical sounds,” Ph.D. dissertation, University of Copenhagen, Denmark, 1999.

- [46] M. Slaney, “Auditory toolbox version 2,” Interval Research Corporation, Tech. Rep., 1998.
- [47] P. Masri, “Computer modelling of sound for transformation and synthesis of musical signals.” Ph.D. dissertation, University of Bristol, United Kingdom, 1996.
- [48] E. Gómez, “Tonal description of music audio signals,” Ph.D. dissertation, Universitat Pompeu Fabra, Barcelona, Spain, 2006.
- [49] C. Harte, M. Sandler, and M. Gasser, “Detecting harmonic change in musical audio,” in *Proceedings of the 1st ACM workshop on Audio and music computing multimedia*. ACM, 2006, pp. 21–26.
- [50] R. Begleiter, R. El-Yaniv, and G. Yona, “On prediction using variable order Markov models,” *Journal of Artificial Intelligence Research*, vol. 22, pp. 385–421, 2004.
- [51] B. Efron and B. Efron, *The jackknife, the bootstrap and other resampling plans*. SIAM, 1982, vol. 38.
- [52] B. Sturm, “Two systems for automatic music genre recognition: What are they really recognizing?” in *Proc. 2nd ACM Intern. workshop on Music information retrieval with user-centered and multimodal strategies*, 2012, pp. 69–74.
- [53] J. Ollen, “A criterion-related validity test of selected indicators of musical sophistication using expert ratings,” Ph.D. dissertation, Ohio State University, United States of America, 2006.
- [54] J. Serrà, H. Kantz, X. Serra, and R. Andrzejak, “Predictability of music descriptor time series and its application to cover song detection,” *IEEE Trans. Audio, Speech and Language Process.*, vol. 20, no. 2, pp. 514–525, 2012.
- [55] F. Takens, “Detecting strange attractors in turbulence,” *Dynamical Systems and Turbulence*, pp. 366–381, 1981.
- [56] P. Foster, S. Dixon, and A. Klapuri, “Identification of cover songs using information theoretic measures of similarity,” in *Proc. IEEE Intern. Conf. Acoustics, Speech and Signal Process. (ICASSP)*, 2013.
- [57] J. Pinto da Costa, H. Alonso, and J. Cardoso, “The unimodal model for the classification of ordinal data,” *Neural Networks*, vol. 21, no. 1, pp. 78–91, 2008.
- [58] C. Ding and H. Peng, “Minimum redundancy feature selection from microarray gene expression data,” *Journal of Bioinformatics and Computational Biology*, vol. 3, no. 2, pp. 185–205, 2005.

- [59] H. Peng, F. Long, and C. Ding, “Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005.
- [60] A. Flexer and D. Schnitzer, “Effects of album and artist filters in audio similarity computed for very large music databases,” *Computer Music Journal*, vol. 34, no. 3, pp. 20–28, 2010.
- [61] A. Hoerl and R. Kennard, “Ridge regression: Biased estimation for nonorthogonal problems,” *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.
- [62] G. Kirchgassner, J. Wolters, and U. Hassler, *Introduction to modern time series analysis*. Springer, 2012.
- [63] G. Box, G. Jenkins, and G. Reinsel, *Time series analysis: forecasting and control*. Wiley, 2013.
- [64] C. W. Granger and R. Joyeux, “An introduction to long-memory time series models and fractional differencing,” *Journal of Time Series Analysis*, vol. 1, no. 1, pp. 15–29, 1980.
- [65] M. Effros, “PPM performance with BWT complexity: A new method for lossless data compression,” in *Proc. Data Compression Conf.*, 2000, pp. 203–212.
- [66] P. Hamel, Y. Bengio, and D. Eck, “Building musically-relevant audio features through multiple timescale representations,” in *Proc. 13th Intern. Society for Music Information Retrieval Conf. (ISMIR)*, 2012, pp. 553–558.
- [67] M. Mauch and S. Ewert, “The audio degradation toolbox and its application to robustness evaluation,” in *Proc. 14th Intern. Society for Music Information Retrieval Conf. (ISMIR)*, 2013, pp. 83–88.
- [68] S. Dubnov, “Unified view of prediction and repetition structure in audio signals with application to interest point detection,” *IEEE Trans. Audio, Speech and Language Process.*, vol. 16, no. 2, pp. 327–337, 2008.
- [69] S. Abdallah and M. Plumbley, “Information dynamics: Patterns of expectation and surprise in the perception of music,” *Connection Science*, vol. 21, no. 2-3, pp. 89–117, 2009.
- [70] R. James, C. Ellison, and J. Crutchfield, “Anatomy of a bit: Information in a time series observation,” *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 21, no. 3, p. 037109, 2011.

Feature name	Description
Chroma (Ellis and Poliner)	12-component chromagram based on using phase-derivatives to identify tonal components in spectrum [43].
dynamics.rms	Root mean square of amplitude.
rhythm.temp	Tempo estimate based on selecting peaks from autocorrelated onsets.
rhythm.attack.time	Duration of onset attack phase.
rhythm.attack.slope	Slope of onset attack phase.
spectral.centroid	First moment of magnitude spectrum.
spectral.brightness	Proportion of spectral energy above 1500Hz.
spectral.spread	Second moment of magnitude spectrum.
spectral.skewness	Skewness coefficient of magnitude spectrum.
spectral.kurtosis	Excess kurtosis of magnitude spectrum.
spectral.rolloff95	95th percentile of energy contained in magnitude spectrum.
spectral.rolloff85	85th percentile of energy contained in magnitude spectrum.
spectral.spectentropy	Shannon entropy of magnitude spectrum.
spectral.flatness	Wiener entropy of magnitude spectrum.
spectral.roughness	Average roughness [44] between peak pairs in magnitude spectrum.
spectral.irregularity	Squared amplitude difference between successive partials [45].
spectral.mfcc	12-component MFCCs [46] (excluding energy coefficient).
spectral.dmfcc	First-order differentiated MFCCs.
spectral.ddmfcc	Second-order differentiated MFCCs.
timbre.zerocross	Zero crossing rate.
timbre.spectralflux	Half-wave rectified L1 distance between magnitude spectrum at successive frames [47].
tonal.chromagram.centroid	Centroid of 12-component chromagram.
tonal.keyclarity	Peak correlation of chromagram with key profiles [48].
tonal.mode	Predicted mode after correlating chromagram with key profiles.
tonal.hcdf	Flux of 6-dimensional tonal centroid [49].

Table 1: Summary of evaluated audio features.

Set	Track representation	Descriptor vector components	Distance measure	Number of distances
1.	FCDs	$\lambda \in \{3, 4, 5\}$	Euclidean	4×25
2.	Frame sequence	N/A	Cross-prediction error	25
3.	FMDs	Mean, Std	Euclidean	25
4.	FMDs	Mean, Var	KLD	25
5.	Combine 3, 4			50
6.	Combine 1, 3, 4			150

Table 2: Summary of descriptor combinations evaluated for similarity rating prediction. Third column denotes components included in descriptor vectors. Fifth column lists size of feature space when performing feature selection for KNN.

Set	Track representation	Descriptor vector components	Prediction coeffs.
1.	FMDs	Mean, Std	$21 \times 2 + 4 \times 24$
2.	FCDs	$\lambda \in \{3, 4, 5\}$	$25 \times 4 \times 3$
3.	Combine 1, 2		

Table 3: Summary of descriptor combinations evaluated for song year prediction. Fourth column lists number of covariates in linear model, excluding intercept.

Set	MAE	RMSE
FCDs	9.49 ± 0.097	11.62 ± 0.108
FMDs	8.26 ± 0.091	10.41 ± 0.113
Combined	7.44 ± 0.086	9.51 ± 0.109

Table 4: Summary of song year prediction accuracy, expressed using MAE and RMSE statistics. Standard errors obtained by bootstrap sampling pairs of predicted and observed chart entry dates and computing MAE, RMSE on each sample.